



Применение ИИ к анализу сетевого трафика для решения задач ИБ

Гетьман Александр Игоревич
кандидат физико-математических наук
старший научный сотрудник ИСП РАН им. В.П. Иванникова



Agenda

- Задачи анализа трафика в контексте ИБ
- Формы представления сетевого трафика и признаковые пространства
- Публичные наборы данных и их ограничения
- Usecase: создание межсетевого экрана уровня приложения (WAF)



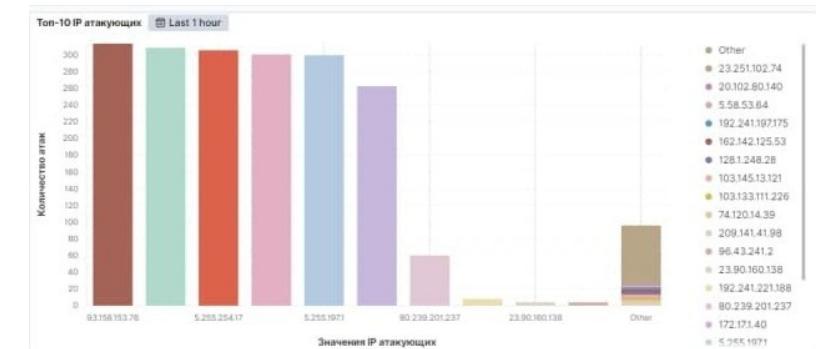
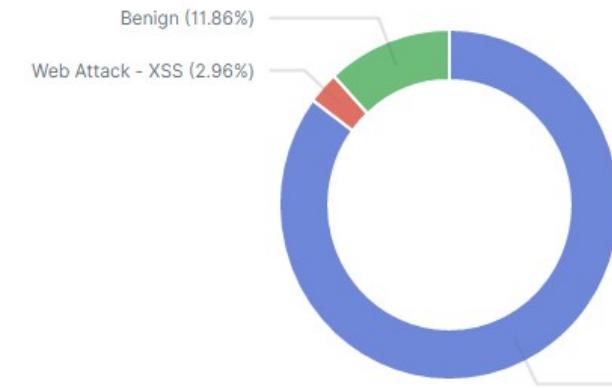
Анализ трафика для задач ИБ

Системы реального времени (онлайн)

- Общее название класса систем — Network Traffic Analysis (NTA)
- Межсетевые экраны уровня приложений (LDAP Firewall, WAF)
- Ограничение доступа к фишинговым ресурсам (AntiPhishing)
- Контроль соблюдения политик ИБ (NSPM)
- Выявление аномалий в трафике (Anomaly Detection)
- Обнаружение DoS и DDoS-атак (DDoS Protection)
- Выявление и классификация вредоносного ПО (Malware classification)
- Обнаружение и предотвращение сетевых атак (IDS/IPS, NGFW, UTM)
- Обнаружение бот-сетей (C&C, AntiBot)

Системы отложенного анализа (оффлайн)

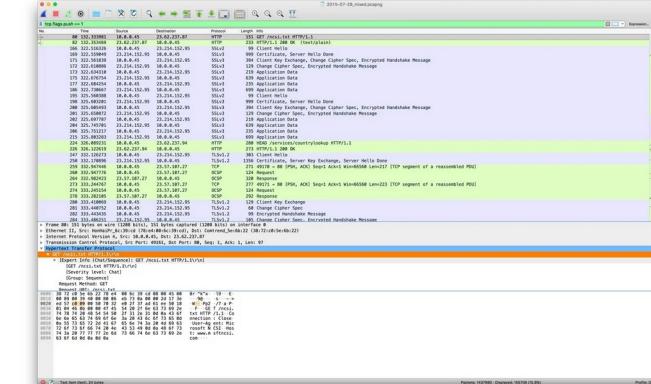
- Обнаружение уязвимостей сетевых сервисов (нмар)
- Поиск угроз (Threat Intelligence, Threat Hunting)
- Расследование инцидентов (Forensics)
- Предотвращение утечек данных (DLP-системы).





Форма представления сетевого трафика

- Для записи данных передаваемых по сети используются специализированное ПО — снiffeры (tcpdump, Wireshark и т.д.)
- Сетевой трафик сохраняется в виде дампа сетевых пакетов с метками времени в форматах PCAP, PCAPNG
- Для просмотра структуры сетевых пакетов (уровни, протоколы, поля, их значение) обычно используют Wireshark
- Смысл имеют не сетевые пакеты по отдельности, а последовательности сетевых пакетов, которыми обмениваются клиенты и серверы в рамках отдельных соединений
- Такие последовательности называются потоками транспортного уровня. Они характеризуются 5-кими (5-tuple) полей (значения этих полей у всех пакетов потока совпадают):
<IP адрес источника, IP-адрес назначения, порт источника, порт назначения, транспортный протокол>
- При этом возможны несколько логических соединений в рамках одного транспортного потока (HTTP/2, QUIC). Также возможно одновременное использование нескольких транспортных потоков одним приложением (браузеры открывают ~5-8 потоков)





Дамп сетевого трафика

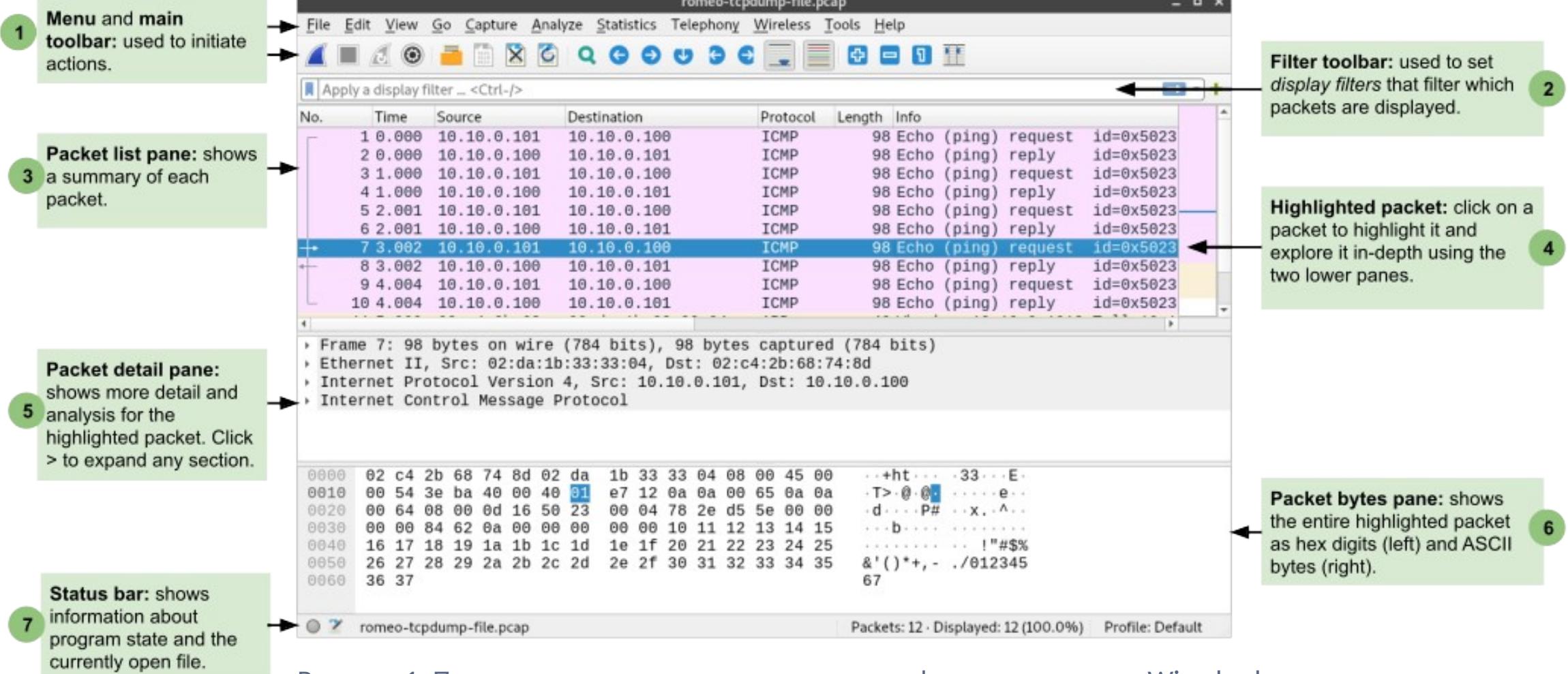


Рисунок 1. Пример просмотра дампа сетевого трафика в программе Wireshark.



Формулировки ML-задачи для анализа трафика

ML-Задача:

- бинарной классификации (атака или нет)
- многоклассовой классификации (вид сетевой атаки)
- поиска аномалий (отклонение от «нормы»)

Входные данные:

Сетевой трафик разбивается на объекты
Выбирается пространство признаков
Объекты отображаются в пространство
признаков и представляются векторами в
этом пространстве

• Обучение:

- **С учителем.** На вход подаются векторы описывающие объект и метки классов объекта
 - Все классы должны быть достаточно равномерно покрыты векторами (проблема дисбаланса классов)
- **Без учителя.** На вход подаются векторы описывающие объект, расстояния между ними, количество кластеров
 - Все классы должны быть достаточно равномерно покрыты векторами
- **Поиск аномалий.** На вход подаются векторы описывающие объект и расстояния между ними
 - Все векторы описывают класс «нормы»



Пространства признаков сетевого трафика

- **Объектом** сетевого трафика, как правило является поток транспортного уровня или его « префикс » при необходимости « ранней классификации » (не более первых N-пакетов, первые 5 секунд и т. д.)
- Для ускорения алгоритма часто упрощают анализ рукопожатий и завершений сеансов TCP-протокола (SYN, FIN, RST), например — по таймеру. Что приводит к проблеммам:
- Один поток → несколько объектов → искаженные значения признаков.
- **Признаки** зависят от задачи
- Задача определяет **набор классов** на которые нужно разбить все объекты
- Признаки должны отражать **релевантные** задаче **характеристики классов**, благодаря которым объекты хорошо разделяются между ними
- Виды **признаковых пространств** трафика:
 - **Признаки пакетов**: значения полей или байтов буфера пакета
 - **Признаки потоков**: размер и длительность потока, количество пакетов, тип протокола, ...
 - **Статистические признаки потока**: средний размер пакета, количество флагов PSH, ...



Наборы данных (датасеты)

- Публичных наборов мало. В зависимости от задачи — от единиц до пары десятков
- Размеры наборов **достаточно маленькие** — десятки-сотни тысяч потоков
- Данные сетевого трафика считаются «**чувствительными данными**», т. к. могут содержать **конфиденциальную и приватную информацию**: имена пользователей, пароли, поисковые запросы, промпты, фотографии документов и т. д.
- Получение хорошего размеченного датасета достаточного размера — **технически сложная, трудоёмкая и затратная по времени задача**
- Для решения проблемы «чувствительных данных», публикуемые датасеты часто содержат не исходные дампы сетевых пакетов, а **вектора признаков (обычно в формате CSV)**, в которые отображены предварительно выделенные объекты сетевого трафика. Недостаток этого формата:
 - Невозможно проверить и верифицировать данные
 - Невозможно изменить признаковое пространство
 - Невозможно пересчитать значения признаков, если в инструменте извлечения признаков выявляется ошибка
 - Затруднительно использовать «смесь» из нескольких датасетов



Влияние характеристик датасетов и условий применения на ML-модели

- Векторы достаточно низкой размерности (десятки-сотни параметров)
- Количество векторов достаточно невелико (десятки-сотни тысяч)
- Онлайн анализ — очень высокие требования к пропускной способности ($10 \text{ Гб/с} \rightarrow 14,88 \text{ Mpps} \rightarrow \sim 1 \text{ Mfps}$)
- Трафик меняет характеристики — требуется до(пере)обучение
- Как следствие — используются в основном не DL-модели (нейросети), а ML-модели (SVM, DT, RF)
- Ниже требования к аппаратуре для обучения и вывода, лучшие метрики качества, выше скорость обучения и пропускная способность
- Чаще всего используют статистические признаки

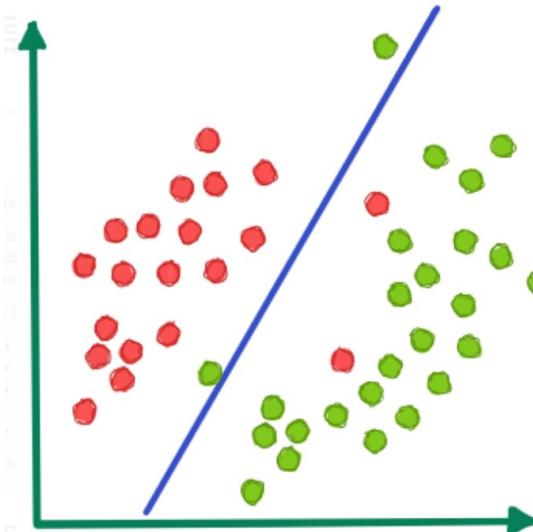
| № | Признак |
|-----|----------------------------|
| 1. | Flow Duration |
| 2. | Total Fwd Packet |
| 3. | Total Bwd Packets |
| 4. | Total Length of Fwd Packet |
| 5. | Total Length of Bwd Packet |
| 6. | Fwd Packet Length Min |
| 7. | Fwd Packet Length Max |
| 8. | Fwd Packet Length Mean |
| 9. | Fwd Packet Length Std |
| 10. | Bwd Packet Length Min |
| 11. | Bwd Packet Length Max |
| 12. | Bwd Packet Length Mean |
| 13. | Bwd Packet Length Std |
| 14. | Flow Bytes/s |
| 15. | Flow Packets/s |
| 16. | Flow IAT Mean |
| 17. | Flow IAT Std |



Задачи классификации и поиска аномалий

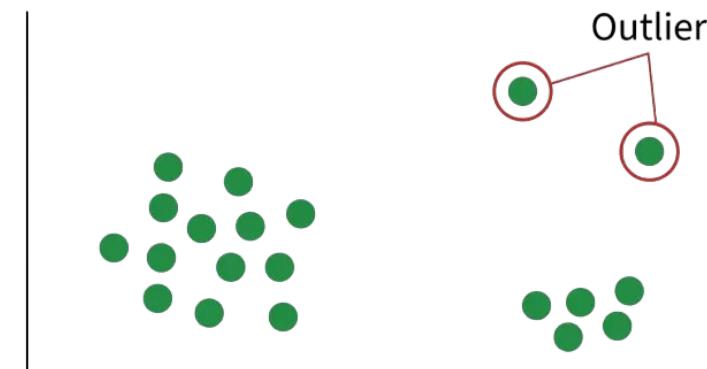
Классификация

- Относительно небольшие размеры наборов данных: сложно генерировать, размечать, чувствительная информация.
- Применение классических ML-алгоритмов (не DL)
 - SVM, DT, RF.



Поиск аномалий

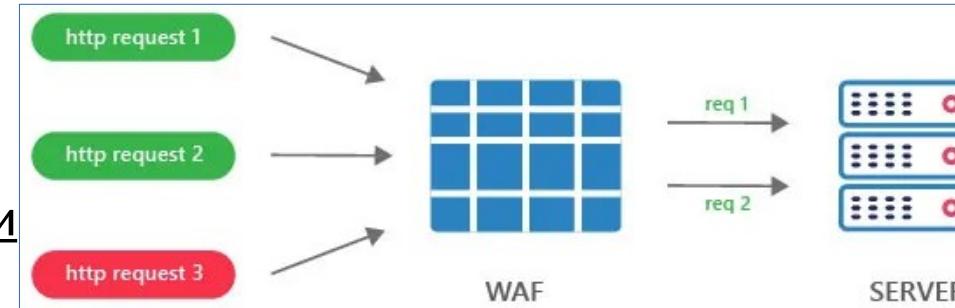
- Аномалия (выброс, outlier) — наличие (небольших) отклонений от некоторой «нормы».
- Необходимость использования только «нормального» трафика. На практике доступен «серый» (смесь нормы и атак)
- Применяемые алгоритмы — Isolation Forest (IF), AutoEncoder (AE), Local Outlier Factor (LOF).



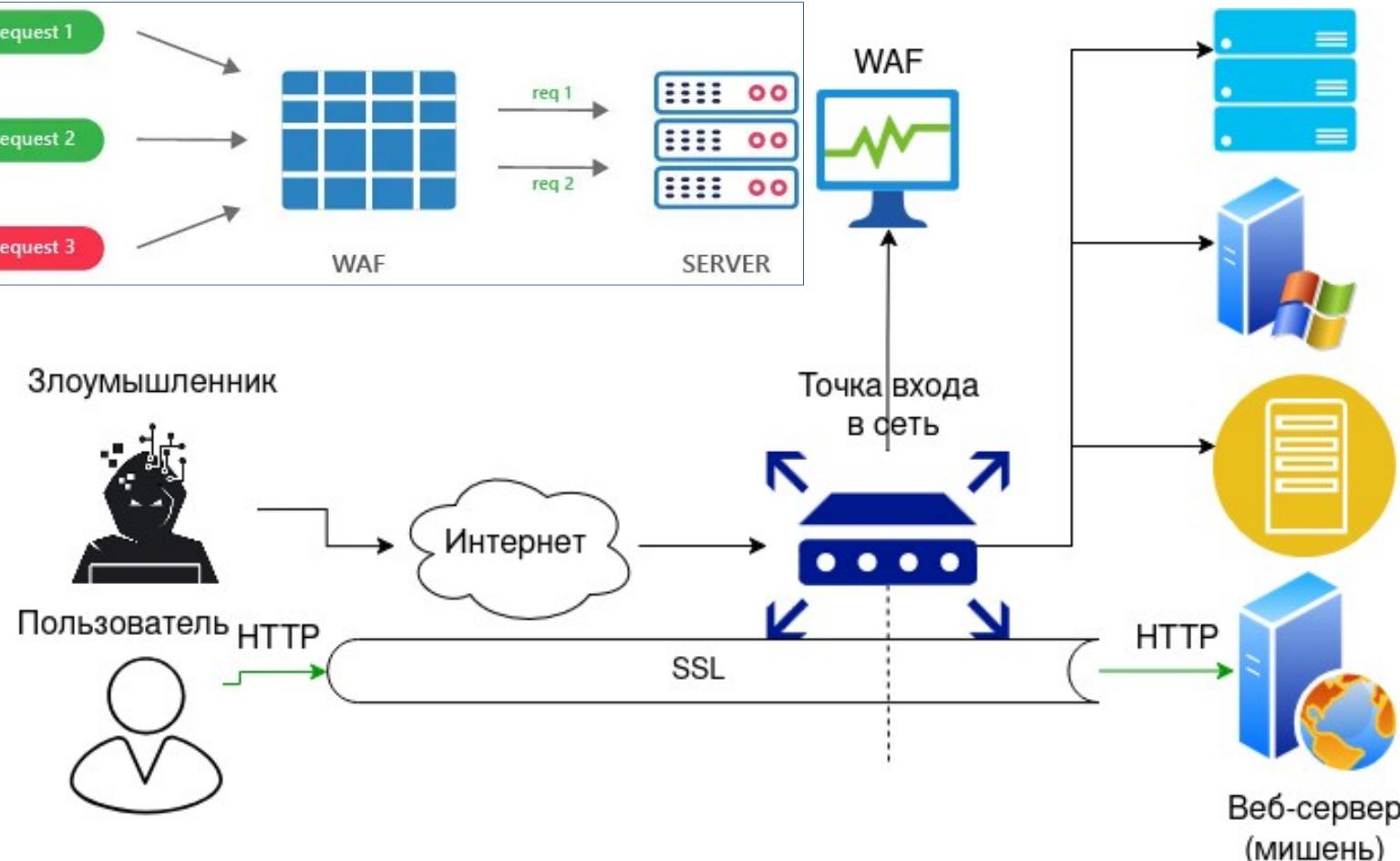


Use case: создание WAF на основе ИИ

В точке наблюдения (**периметр сети**) виден весь трафик между внешними пользователями и внутренними системами, но не между внутренними пользователями и внутренними системами



- Значительная часть трафика, особенно Web (HTTP 1-3) – зашифрована
- Часто появляются новые атаки (Zero-day)
- Нужна защита до исправления уязвимости и создания сигнатур
- Первая задача – получение данных для обучения:
 1. Публичный размеченный датасет
 2. Создание собственного датасета





Проблемы публичных датасетов (1)



- Отсутствует стандартный инструмент разметки и набор признаков
- Привязаны к инструменту извлечения признаков
 - инструменты не заточены на работу онлайн
 - могут содержать ошибки в коде извлечения
 - несовместимы между собой:
 - разный набор признаков
 - разный формат представления (string vs int)
 - разная точность данных (mc vs nc).
- **CICFlowMeter** – один из наиболее популярных инструментов.
 - Разработан Canadian Institute for Cybersecurity
 - Найдено более 5 типов ошибок.
 - Разработаны скорректированные инструменты, данные пересчитаны.

| № | Название инструмента | Поддержка платформы | Язык программирования | Количество выделяемых признаков | Известные наборы данных |
|---|----------------------|-------------------------------------|-----------------------|--|-------------------------|
| 1 | FCParser | Unix | Python | Переменное количество (методология FaaS) | UGR-16 |
| 2 | MADAM ID | Сведения отсутствуют | Сведения отсутствуют | Сведения отсутствуют | KDD Cup 1999 |
| 3 | Argus | Windows, Linux, Solaris, OS X и др. | C | 125 | CTU-13, UNSW-NB15 |
| 4 | NFStreams | Linux, MacOS, ARM | Python | 48 | Сведения отсутствуют |
| 5 | CICFlowMeter | Linux | Java/C | 85 / 80 (разные версии) | CICIDS 2017-2019 |
| 6 | LycosTand | Linux | C | 82 | LYCOS-IDS2017 |



Проблемы публичных датасетов (2)

- Эксперимент с обучением на публичном наборе и применением на реальной сети
- Анализ причин показал:
 - неадекватность применяемых средств атак;
 - **сильное влияние характеристик стенда (пропускная способность сети, настройки ПО и т.д.).**
- Возможные решения:
 - генерация данных на целевой сети;
 - **оценка возможности переноса данных.**

| | Запуск 1 | Запуск 2 |
|----------------------------|--|--|
| Обучение модели | | |
| Набор данных | Сбалансированное подмножество веб-атак CICIDS2017 | |
| Набор для обучения | 70% сбалансированного подмножества веб-атак CICIDS2017 | |
| Модель ИИ | RandomForestClassifier | |
| Множество признаков | <ol style="list-style-type: none">1. Average Packet Size2. Flow Bytes/s3. Max Packet Length4. Fwd Packet Length Mean5. Fwd IAT Min6. Total Length of Fwd Packets7. Fwd IAT Std8. Flow IAT Mean9. Fwd Packet Length Max10. Fwd Header Length | |
| Тестирование модели | | |
| Тестовый набор | 30% сбалансированного подмн-ва веб-атак CICIDS2017 | 100%, сбалансированного набора трафика реальной сети |
| Метрики | | |
| Accuracy | 0.983 | 0.456 |
| Precision | 0.982 | 0.812 |
| Recall | 0.961 | 0.033 |
| F1 | 0.971 | 0.064 |



Получение релевантных данных

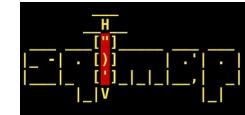
Публичных данных недостаточно

- их релевантность вызывает вопросы
- не факт что эти данные актуальны для вашего сетевого окружения (сервисы, пропускная способность, сетевые настройки)

Необходимо получить данные на своей сети

- Нормальный пользовательский “фоновый” трафик
- Трафик сетевых атак
- Источник обоих видов трафика должен быть вне внутренней сети

- Данных необходимо достаточно много
- Нужен стенд для их генерации
- Нужен трафик 2x видов: нормальный и тех web-атак, которые планируется детектировать
- Пользовательский web-трафик – это в основном браузер (а также curl, wget и т.д.)
- Наиболее распространённые web-атаки: XSS, CSRF, SQL Injection, Comand Injection, Web Shell, Brute force
- Вручную запускать достаточно трудоёмко и долго
- Для автоматизации браузера существует Selenium
- Для генерации атак можно использовать специализированный дистрибутив пентестеров Kali Linux с набором инструментов атаки (xsser, weevly, commix, sqlmap, patator) и скрипты их автоматизации





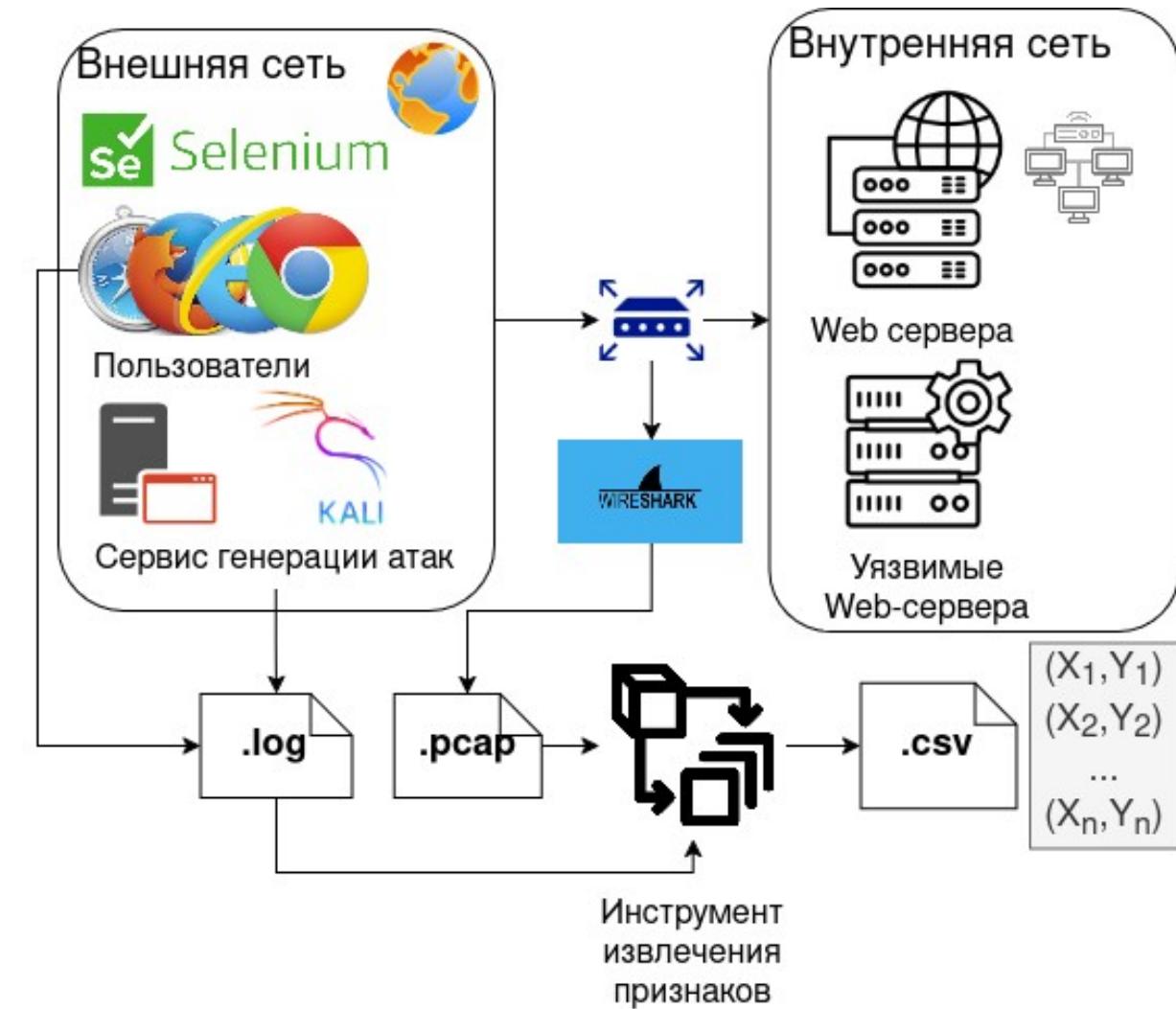
Создание стенда

Онлайн-фаза

1. Запуск в т.ч. уязвимых веб серверов DVWA, XVWA, OWASP WebGoat, NodeGoat, Juice Shop, Mutillidae II
2. Запуск перехвата сетевого трафика
3. Запуск генераторов нормального трафика и трафика атак с журналом создаваемых соединений

Оффлайн-фаза

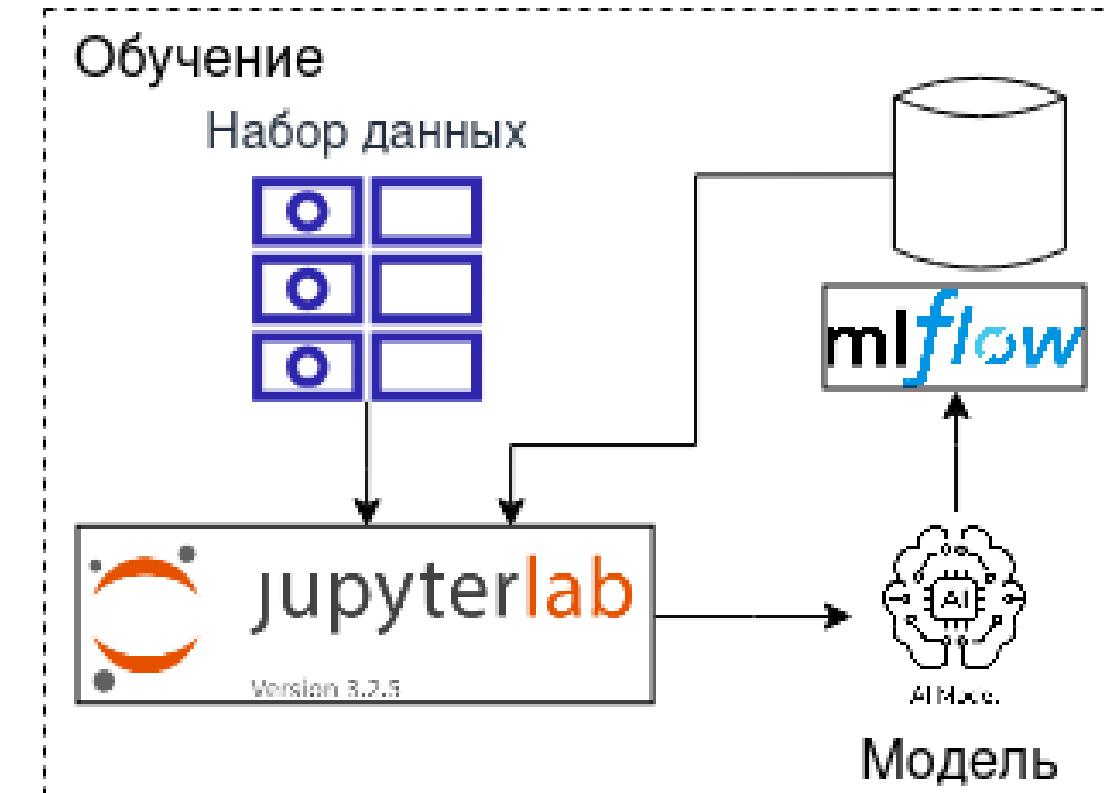
1. Использование инструмента извлечения признаков и разметки для генерации датасета





Создание модели (MLOps)

- Анализ и выбор релевантных признаков
- Выбор подмножества наиболее ценных признаков (сокращение размерности признакового пространства)
- Разбиение данных на обучающую, валидационную и тестовую выборки
- Выбор оптимальной модели
- Подбор оптимальных гиперпараметров
- Оценка качества работы модели
- Сохранение «артефактов» модели и самой модели в реестре моделей





Оценка влияния загрязнения данных

Гипотеза:

- Недостаточный анализ особенностей выбранных признаков может сильно ухудшать результаты

Реализация стенда:

- 1) На этапе формирования web-сервер вставляет флаг URG во все исходящие пакеты TCP.
- 2) На этапе эксплуатации ML модели данный флаг не устанавливается.

| Качество модели на загрязнённом наборе данных (Fwd Flag URG = 1) | Качество модели на реальном трафике (Fwd Flag URG = 0) |
|---|---|
| Accuracy = 0.999 Precision = 0.999 Recall = 0.999 F1 = 0.999 | Accuracy = 0.718 Precision = 1.0 Recall = 0.146 F1 = 0.254 |

Подходы к решению проблемы:

- Необходима оценка дисперсии признаков в пределах классов.
- Выявленные признаки низкой дисперсии могут рассматриваться как признак загрязнения.



Оценка влияния характеристик сети

Предпосылки:

- В условиях шифрования трафика одна из наиболее информативных групп признаков для классификации — временные признаки (интервалы между пакетами).
- Данная характеристика сильно зависит от характеристик целевой сети, её текущей загруженности и доли потерь.

Гипотеза:

- Изменение характеристик сети с момента обучения сильно влияет на точность ML-модели

| Величина вносимой задержки (мс) | Точность классификации (Accuracy) |
|---------------------------------|-----------------------------------|
| 0 | 0.97 |
| 900 | 0.873 |
| 1100 | 0.924 |
| 1300 | 0.858 |
| 1700 | 0.923 |

Реализация стенда:

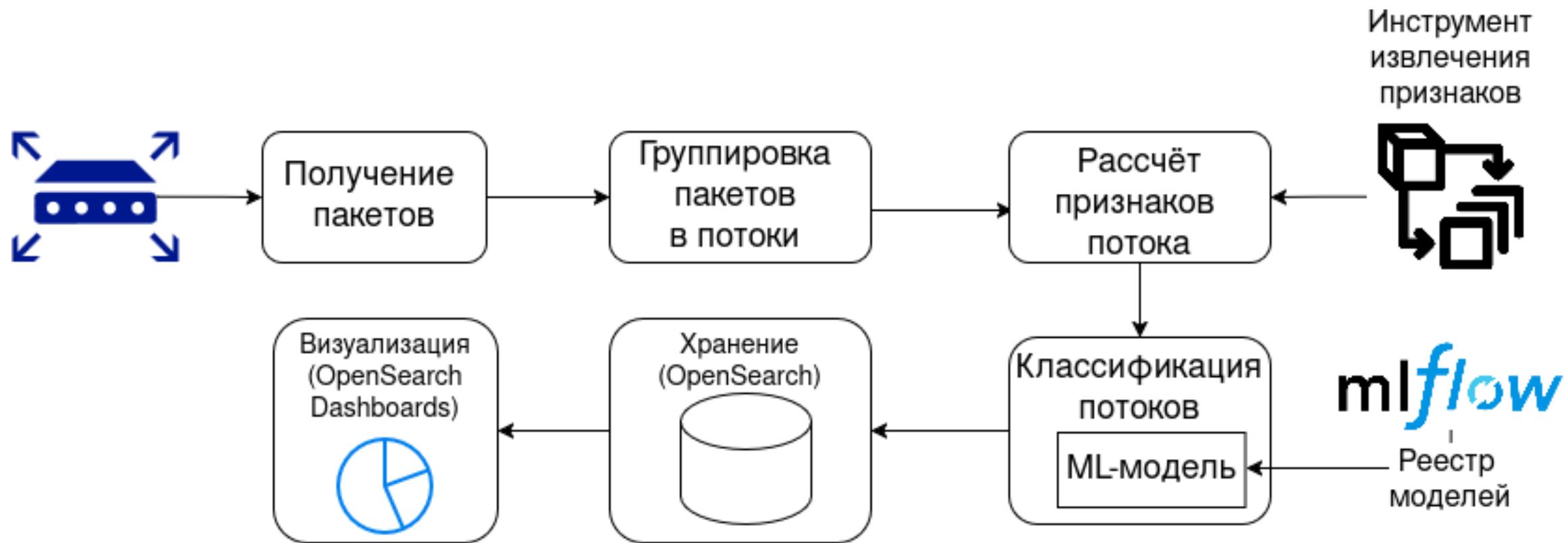
- Внесение временных задержек отправкой отдельных пакетов с помощью программного модуля ядра `netem`.

Подходы к решению проблемы:

- Добавление к обучающему набору примеров потоков с замедлением в некотором диапазоне



Формирование конвеера обработки трафика в реальном времени





Оценка возможности обнаружения

Zero-day атак

Эмуляция 0-day уязвимости:

- Набор сигнатур обнаружения атак OWASP ModSecurity Core Rule Set v3.3.2, который используется в **WAF ModSecurity**
- При обучении модели данный набор правил и соответствующих атак не использовался
- Найдено сообщение об ошибке в базе сигнатур "**Drop keyword not blocked for SQL injection**":
 - для **SQL** инструкции **DROP** отсутствовал шаблон поиска в правиле **942360** и запрос вида:
https://some.web.site/index.html?q='drop table users;--
не блокировался до обновления набора правил
- Данное правило удалено
- Смоделирована ситуация, 0-day уязвимости, т.к. в базе правил отсутствует информация о данной SQL инъекции.



Результат оценки

- Отправка соответствующего SQL-запроса (**SQL Injection**) не детектируется сигнатурной **WAF ModSecurity**, но детектируется **WAF на основе ИИ**
- WAF на основе ИИ способен детектировать Zero-day атаки благодаря обобщающей способности



Поддержание модели в актуальном состоянии. Дрейф данных

Дрейф данных - изменение характеристик входных данных со временем, приводящее к снижению качества работы модели.

Виды дрейфа:

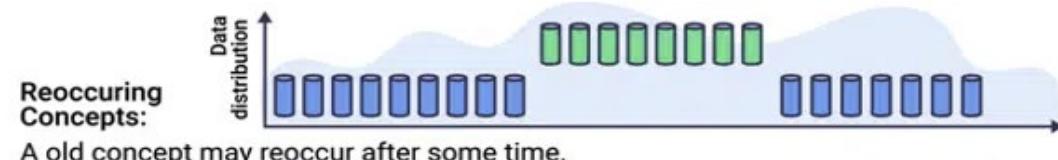
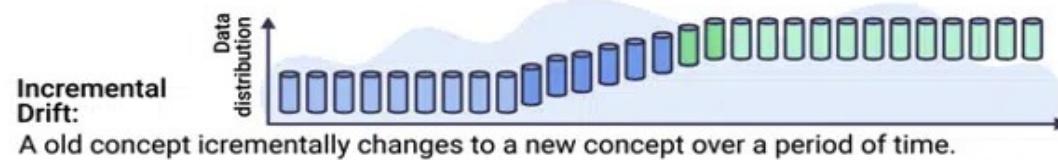
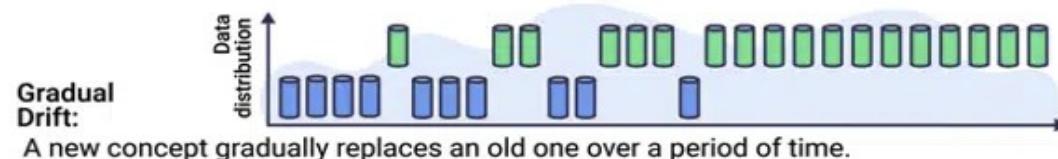
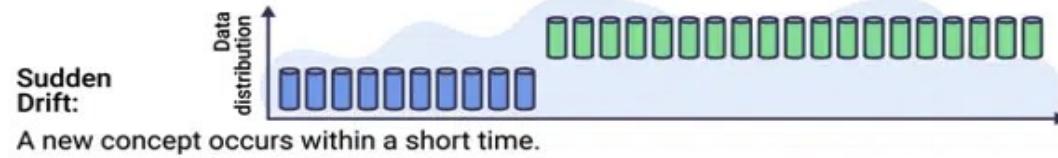
- дрейф концепции (Concept drift);
- дрейф данных (Data drift, Feature drift);
- дрейф предсказания (Prediction drift);
- дрейф метки (Label drift).

Подходы к обнаружению:

- в потоковых данных;
- в накопленных данных.

Причины дрейфа в случае анализа трафика:

- Новые приложения и протоколы, изменение характеристик сети, настроек приложений, и т.д.





Оценка влияния дрейфа данных на точность модели

Объект защиты

Web-приложение Damn Vulnerable Web Application (DVWA) запущенное на веб-сервере Apache

Конфигурация веб-сервера на которой происходил сбор данных для обучения (конфигурация 1)

- KeepAliveTimeout = 5
- MaxKeepAliveRequests = 100

Конфигурация 2

- KeepAliveTimeout = 65
- MaxKeepAliveRequests = 1000

| Тип сетевой атаки | Точность классификации (Accuracy) Конфигурация 1 | Точность классификации (Accuracy) Конфигурация 2 |
|------------------------|---|---|
| Sql Injection (Sqlmap) | 0.989 | 0.635 |
| Brute Force (Patator) | 0.709 | 0.312 |

Анализ

Увеличение параметра KeepAliveTimeout приводит к увеличению запросов, попадающих в одну сессию и, как следствие, к увеличению продолжительности сессий, что влияет на многие признаки

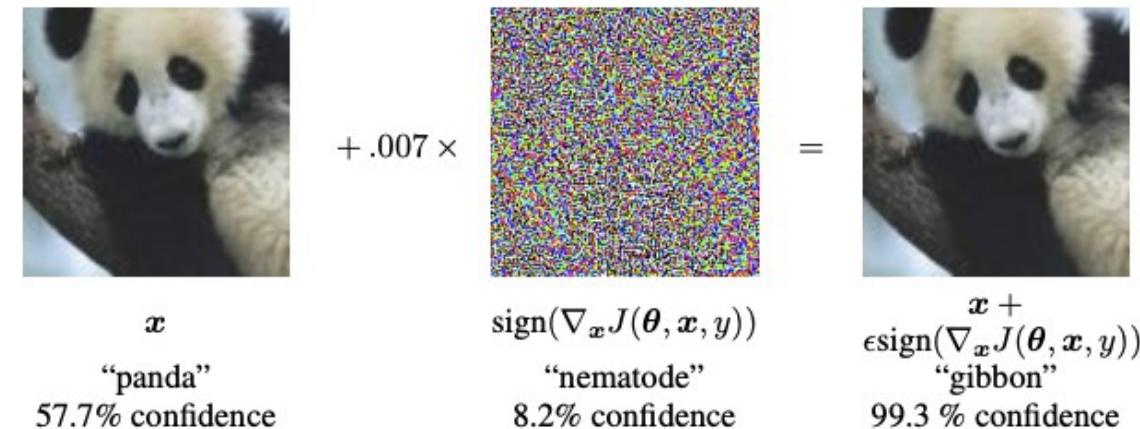
Вывод

Необходимо детектировать дрейф данных и проводить до(пере)обучение модели



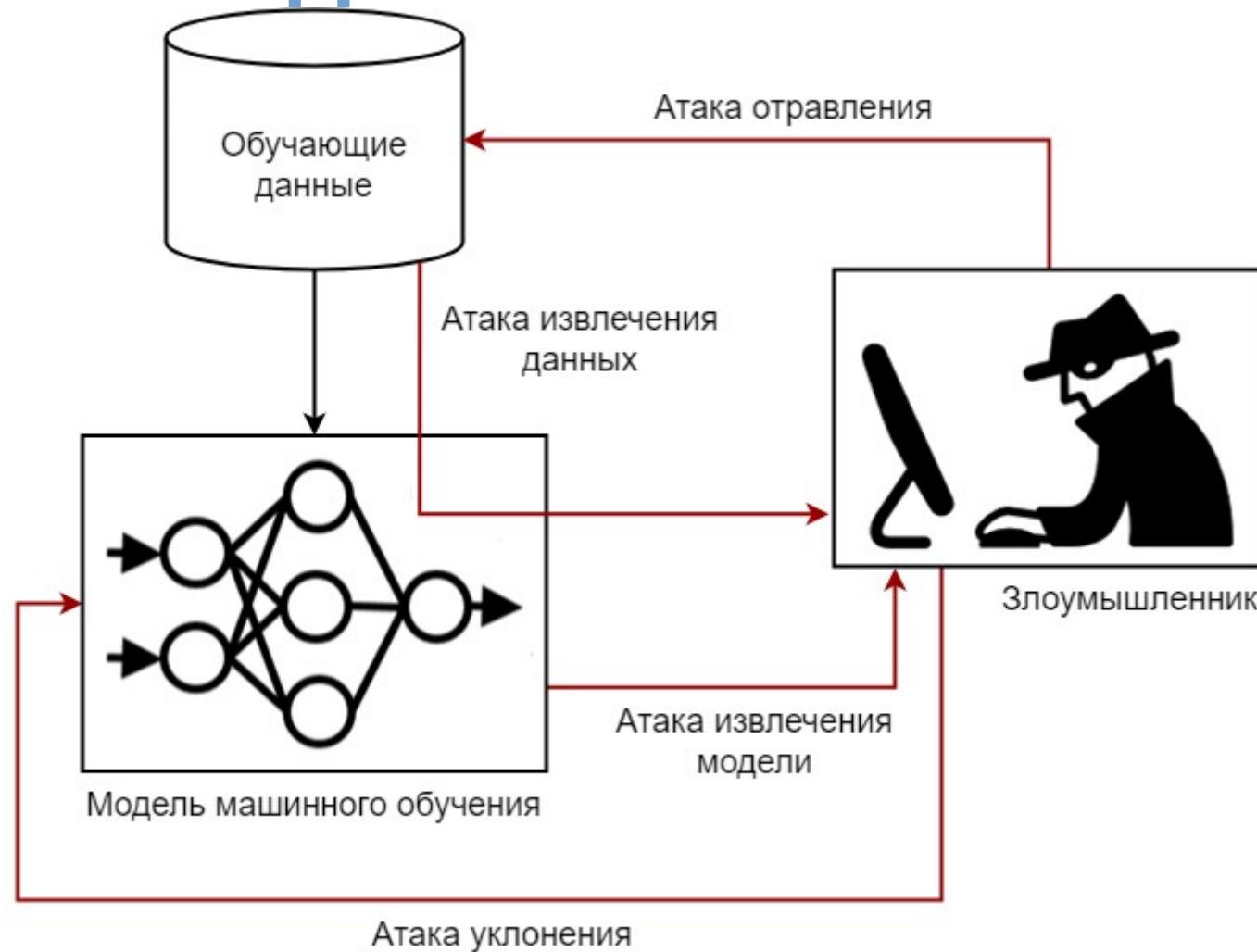
Применение ИИ несёт и дополнительные риски. Состязательные атаки

- Состязательная атака типа уклонения - **незначительное** изменение вектора признаков, которое приводит к некорректной классификации объекта.
- Основная проблема – возможность автоматической генерации изменённых векторов – т.н. **«состязательных примеров» (СП)**
- Атаки делятся на виды – чёрный (ЧЯ) и белый ящик (БЯ), в зависимости от того нужен ли доступ к модели для генерации состязательных примеров
- Существуют программные платформы, для автоматической генерации, например ART (Adversarial Robustness Toolbox)
- Применимость алгоритмов белого ящика зависят от конкретного типа модели, например нейросети (НС) или решающего дерева (РД)





Основные виды состязательных атак





Алгоритмы состязательных атак

| Метод генерации (состязательная атака) | Год | Класс атаки | Целевая модель | Пример реализации |
|--|------|-------------|----------------|---|
| FGSM – Fast Gradient Signed Method | 2015 | БЯ | НС | Adversarial Robustness Toolbox (ART), FastGradientMethod |
| MILP – Mixed-integer Linear Programming | 2015 | БЯ | РД | https://github.com/YihanWang617/On-ell_p-Robustness-of-Ensemble-Stumps-and-Trees |
| JSMA – Jacobian-based Saliency Map Attack | 2016 | БЯ | НС | ART, SaliencyMapMethod |
| BIM – Basic Iterative Method | 2017 | БЯ / ЧЯ | НС | ART, BasicIterativeMethod |
| CW – Carlini and Wagner attack | 2017 | БЯ / ЧЯ | НС | ART, CarliniL[X]Method |
| PGD – Projected gradient descent | 2017 | БЯ / ЧЯ | НС | ART, ProjectedGradientDescent |
| ZOO – Zeroth order optimization based black-box attack | 2017 | ЧЯ | НС | ART, ZooAttack |
| MIM – Momentum Iterative Method | 2018 | БЯ / ЧЯ | НС | CleverHans library, momentum_iterative_method |
| The Cube Attack | 2019 | БЯ | РД | https://github.com/max-andr/provably-robust-boosting |
| RBA – Region-Based Attack | 2019 | БЯ | РД | https://github.com/chenhongge/RobustTrees |
| HSJA – Hop Skip Jump Attack | 2019 | ЧЯ | НС | ART, HopSkipJump |
| Sign-OPT | 2020 | ЧЯ | НС | ART, SignOPTAttack |
| LT – Leaf Tuple | 2020 | БЯ | РД | https://github.com/chong-z/tree-ensemble-attack |



Подходы к защите от состязательных атак

- «Размытие границ» принятия решений моделью – внесение в обучающую выборку примеров на основе исходных с добавлением некоторого шума (например гауссовского)
- Обнаружение состязательных атак, как векторов, близких к границе принятия решения
- Удаление признаков – выявление наиболее уязвимых признаков и удаление их из обучающих данных
- Состязательное обучение – генерация состязательных примеров и их добавление в обучающий набор с корректной разметкой класса
- Использование ансамблей из нескольких моделей ИИ



Thank you!

Questions?

Contact: ever@ispras.ru